**MODULE: 5 (DATABASE)**

* What do you understand By Database

Answer: DBMS stands for Data Base Management System.

Data + Management System

Database is a collection of inter-related data and Management System is a set of programs to store and retrieve those data.

DBMS is a collection of inter-related data and set of programs to store & access those data in an easy and effective manner.

* What is Normalization ?

Answer: Normalization is the process of minimizing redundancy (duplicity) from a relation or set of relations.

Redundancy in relation may cause insertion, deletion and updation anomalies. So, it helps to minimize the redundancy in relations.

Most Commonly used normal forms:

1. First normal form
2. Second normal form
3. Third normal form

* What is Difference between DBMS and RDBMS?

Answer:

|  |  |
| --- | --- |
| DBMS | RDBMS |
| DBMS stands for “Database Management System.” | RDBMS stands for “Relational Database Management System.” |
| DBMS technology stores the data in the form files. | RDBMS stores the data in the form of tables. |
| DBMS is designed to handle small amounts of data. | RDBMS is designed to deal with vast amount of data. |
| DBMS provides support only for a single user at a time. | RDBMS provides support for multiple users at a time. |

* What is MF Cod Rule of RDBMS Systems?

Answer:

Rule 1 : The information rule. All information in the database should be represented in the same way, and stored in tables in the form of rows and columns.

Rule 2 : The guaranteed access rule. All data must be accessible logically using the table name, primary key and column.

Rule 3 : Systematic treatment of null values. The Null values in a database should be handled systematically and uniformly.

Rule 4 : Active online catalog. The structure of the entire database must be stored in an online catalog, as a data dictionary.

Rule 5 : The comprehensive data sub language rule. The system must support at least one relational language that 1. Has a linear syntax 2. Can be used both interactively and within application programs 3. Supports data definition operations, data manipulation operations, security and integrity constraints, and transaction management operations.

Rule 6 : The view updating rule. All views that can be updated theoretically, must be updated by the system.

Rule 7 : High-level insert, update, and delete. The system must support insert, update, and delete operations on the database. It should also support operators that manipulate a set of rows instead of just a single row.

Rule 8 : Physical data independence. Data stored in the database should be independent of how it is being accessed by external applications.

Rule 9 : Logical data independence. Any changes in the logical data should not impact the applications using it.

Rule 10 : Integrity independence. Integrity constraints must stay separate from the application using it. These constraints must be specified separately and cataloged.

Rule 11 : Distribution independence. The distribution of data to different servers and locations should be hidden from the user.

Rule 12 : The non-subversion rule. The database system access should never bypass a relational security or integrity constraint.

* What do you understand By Data Redundancy?

Answer: Data Redundancy means the occurrence of duplicate copies of similar data. It is done intentionally to keep the same piece of data at different places, or it occurs accidentally.

* What is DDL Interpreter?

Answer: DDL Interpreter DDL expands to Data Definition Language. DDL Interpreter as the name suggests interprets the DDL statements such as schema definition statements like create, delete, etc. The result of this interpretation is a set of a table that contains the meta-data which is stored in the data dictionary.

* What is DML Compiler in SQL?

Answer: DML Compiler DML expands to Data Manipulation Language.

A Data Manipulation Language is a computer programming language used for inserting, deleting, and modifying data in a   
database. A DML is often a sublanguage of a broader database language such as SQL, with the DML comprising some of the operators in the language.

* What is SQL Key Constraints writing an Example of SQL Key Constraints?

Answer: SQL Key Constraints are used to enforce the integrity and uniqueness of data in a relational database. There are several types of key constraints in SQL:

1. Primary Key Constraint: A Primary Key is a column that uniquely identifies each row in a table. It enforces the uniqueness of values in the specified column and ensures that the values are not NULL. A table can have only one primary key.

Example of defining a primary key constraint in SQL:

create table employee(

id int primary key,

name text not null,

address varchar(30),

phone\_no varchar(10) unique

);

1. Unique Key Constraint: A unique key constraint ensures that the values in the specified column are unique, but it allows NULL values. Unlike a primary key, a table can have multiple unique key constraints.

Example of defining a unique key constraint in SQL:

create table Products(

product\_id int,

product\_name varchar(100) unique);

1. Foreign Key Constraint: A Foreign Key Constraint establishes a relationship between two tables by linking a column in one table to the primary key in another table. It ensures that values in the column with the foreign key match the values in the referenced primary key column.

Example of defining a foreign key constraint in SQL:

create table Exam(

Rollno int,

S\_code text not null,

Marks int not null,

P\_code text not null,

foreign key(Rollno) references Student(Rollno)

);

* What is save Point? How to create a save Point write a Query?

Answer: A SAVEPOINT is a point in a transaction when you can roll the transaction back to a certain point without rolling back the entire transaction.

The syntax for a SAVEPOINT command is as shown below:

SAVEPOINT SAVEPOINT\_NAME;

* What is trigger and how to create a Trigger in SQL?

Answer: An SQL trigger is a database object that is associated with a table and automatically executes a set of sql statements when a specific event occurs on that table.

Before insert trigger :

create trigger trigger\_name

before insert on table\_name

for each row

-- trigger action code goes here

After insert trigger :

create trigger trigger\_name

after insert on table\_name

for each row

-- trigger action code goes here